Unitary synthesis with fewer T gates

Xinyu (Norah) Tan

Google Quantum AI
MIT

TQC 2026, August 27

arXiv: 2509.25702

Unitary synthesis

- Decompose arbitrary unitaries into smaller, structured circuits
- Exact synthesis with continuous gate sets
  - e.g. single-qubits + CNOTs: $\Theta(4^n)$
- Discrete gate sets: approximation to arbitrary precision ϵ
  - Solovay-Kitaev theorem: $\text{polylog}(1/\epsilon)$
- Clifford + T circuits: fault-tolerance
  - Clifford is cheap, $\mathrm{T} = R_z(\pi/8)$ is expensive
  - Clean ancillae $|0\rangle$ can be cheap too, depending on the architecture (e.g. neutral atom)
  - Non-adaptive (fully unitary) implementation > adaptive circuits

How to decompose arbitrary unitaries using as few T gates as possible?
What is the optimal T-count?

Prior work + this work

Special case 1: single-qubit unitaries

- Every $U \in \mathrm{SU}(2)$ can be implemented to error ϵ using $\Theta(\log(1/\epsilon))$ Hadamard and T gates; no ancilla [Ross & Selinger, 2014]

Take $\epsilon = O(1)$ to focus on the n-dependence; $\tilde{O}$ and $\tilde{\Omega}$ hides $\mathrm{poly}(n)$

	Lower bound		**Upper bounds**		
	#T	#ancilla	#T	#ancilla	
Special case 2: **quantum state preparation**; first column of $U \in \mathrm{U}(2^n)$					

Prior work + this work

Special case 1: single-qubit unitaries

- Every $U \in \mathrm{SU}(2)$ can be implemented to error ϵ using $\Theta(\log(1/\epsilon))$ Hadamard and T gates; no ancilla [Ross & Selinger, 2014]

Take $\epsilon = O(1)$ to focus on the n-dependence; $\tilde{O}$ and $\tilde{\Omega}$ hides poly(n)

	Lower bound		**Upper bounds**		
	#T	#ancilla	#T	#ancilla	
Special case 2: quantum state preparation; first column of $U \in \mathrm{U}(2^n)$	$\tilde{\Omega}(2^n/\lambda)$	λ	$\Theta(2^{n/2})$	$O(2^{n/2})$	[Low, Kliuchnikov, Schaeffer, 2018] [Gosset, Kothari, Wu, 2024]

Lowering T-count must increase ancilla-count

Prior work + this work

Special case 1: single-qubit unitaries

- Every $U \in \mathrm{SU}(2)$ can be implemented to error ϵ using $\Theta(\log(1/\epsilon))$ Hadamard and T gates; no ancilla [Ross & Selinger, 2014]

Take $\epsilon = O(1)$ to focus on the n-dependence; $\tilde{O}$ and $\tilde{\Omega}$ hides poly(n)

	Lower bound		Upper bounds		
	#T	#ancilla	#T	#ancilla	
Special case 2: quantum state preparation; first column of $U \in \mathrm{U}(2^n)$	$\tilde{\Omega}(2^n/\lambda)$	λ	$\Theta(2^{n/2})$	$O(2^{n/2})$	[Low, Kliuchnikov, Schaeffer, 2018] [Gosset, Kothari, Wu, 2024]
General unitary synthesis of $U \in \mathrm{U}(2^n)$	$\tilde{\Omega}(2^{2n}/\lambda)$	λ			
	$\Omega(2^n)$ arbitrary ancillae and adaptivity [GKW24]				

Lowering T-count must increase ancilla-count

Prior work + this work

Special case 1: single-qubit unitaries

- Every $U \in \mathrm{SU}(2)$ can be implemented to error ϵ using $\Theta(\log(1/\epsilon))$ Hadamard and T gates; no ancilla [Ross & Selinger, 2014]

Take $\epsilon = O(1)$ to focus on the n-dependence; $\tilde{O}$ and $\tilde{\Omega}$ hides $\mathrm{poly}(n)$

	Lower bound		Upper bounds		
	#T	#ancilla	#T	#ancilla	
Special case 2: quantum state preparation; first column of $U \in \mathrm{U}(2^n)$	$\tilde{\Omega}(2^n/\lambda)$	λ	$\Theta(2^{n/2})$	$O(2^{n/2})$	[Low, Kliuchnikov, Schaeffer, 2018] [Gosset, Kothari, Wu, 2024]
General unitary synthesis of $U \in \mathrm{U}(2^n)$	$\tilde{\Omega}(2^{2n}/\lambda)$	λ	$\tilde{O}(2^{3n/2})$	$O(2^{n/2})$	by reducing to preparing 2^n quantum states [LKS18]
	$\Omega(2^n)$ arbitrary ancillae and adaptivity [GKW24]				

Lowering T-count must increase ancilla-count

Prior work + this work

Special case 1: single-qubit unitaries

- Every $U \in \mathrm{SU}(2)$ can be implemented to error ϵ using $\Theta(\log(1/\epsilon))$ Hadamard and T gates; no ancilla [Ross & Selinger, 2014]

Take $\epsilon = O(1)$ to focus on the n-dependence; $\tilde{O}$ and $\tilde{\Omega}$ hides poly(n)

	Lower bound		Upper bounds		
	#T	#ancilla	#T	#ancilla	
Special case 2: quantum state preparation; first column of $U \in \mathrm{U}(2^n)$	$\tilde{\Omega}(2^n/\lambda)$	λ	$\Theta(2^{n/2})$	$O(2^{n/2})$	[Low, Kliuchnikov, Schaeffer, 2018] [Gosset, Kothari, Wu, 2024]
General unitary synthesis of $U \in \mathrm{U}(2^n)$	$\tilde{\Omega}(2^{2n}/\lambda)$	λ	$\tilde{O}(2^{3n/2})$	$O(2^{n/2})$	by reducing to preparing 2^n quantum states [LKS18]
	$\Omega(2^n)$ arbitrary ancillae and adaptivity [GKW24]		$O(2^{3n/2})$	$O(2^n)$	translated from the number of queries to classical Boolean function oracles [Rosenthal, 2021]

Lowering T-count must increase ancilla-count

Prior work + this work

Special case 1: single-qubit unitaries

- Every $U \in \mathrm{SU}(2)$ can be implemented to error ϵ using $\Theta(\log(1/\epsilon))$ Hadamard and T gates; no ancilla [Ross & Selinger, 2014]

Take $\epsilon = O(1)$ to focus on the n-dependence; $\tilde{O}$ and $\tilde{\Omega}$ hides poly(n)

<table>
<tr><th></th><th colspan="2">Lower bound</th><th colspan="3">Upper bounds</th></tr>
<tr><td></td><td>#T</td><td>#ancilla</td><td>#T</td><td>#ancilla</td><td></td></tr>
<tr><td>Special case 2: quantum state preparation; first column of $U \in \mathrm{U}(2^n)$</td><td>$\tilde{\Omega}(2^n/\lambda)$</td><td>$\lambda$</td><td>$\Theta(2^{n/2})$</td><td>$O(2^{n/2})$</td><td>[Low, Kliuchnikov, Schaeffer, 2018]
[Gosset, Kothari, Wu, 2024]</td></tr>
<tr><td rowspan="3">General unitary synthesis of $U \in \mathrm{U}(2^n)$</td><td>$\tilde{\Omega}(2^{2n}/\lambda)$</td><td>$\lambda$</td><td>$\tilde{O}(2^{3n/2})$</td><td>$O(2^{n/2})$</td><td>by reducing to preparing 2^n quantum states [LKS18]</td></tr>
<tr><td colspan="2" rowspan="2">$\Omega(2^n)$
arbitrary ancillae and adaptivity [GKW24]</td><td>$O(2^{3n/2})$</td><td>$O(2^n)$</td><td>translated from the number of queries to classical Boolean function oracles [Rosenthal, 2021]</td></tr>
<tr><td>$\tilde{O}(2^{4n/3})$</td><td>$\tilde{O}(2^{2n/3})$</td><td>This work</td></tr>
</table>

Lowering T-count must increase ancilla-count

The $\tilde{O}(2^{4n/3})$ T-count algorithm

- Break an arbitrary $U \in \mathrm{U}(2^n)$ down to more structured objects
- **Step 1:** recursively apply the cosine-sine (CS) decomposition
- **Step 2:** implement <u>multi-controlled k-qubit unitaries</u>
  - Controlled by $n - k$ qubits, targets k qubits
  - e.g. $\sum_{x \in \{0,1\}^{n-k}} |x\rangle\langle x| \otimes V_x$

Cosine-sine (CS) decomposition

- Any $U \in \mathrm{U}(2^n)$ can be written as

$$U = \left[\begin{array}{c|c} V_1 & \\ \hline & V_2 \end{array}\right] \left[\begin{array}{cccc|cccc} \boxed{\cos(\theta_1)} & & & & \boxed{\sin(\theta_1)} & & & \\ & \cos(\theta_2) & & & & \sin(\theta_2) & & \\ & & \ddots & & & & \ddots & \\ & & & \cos(\theta_{2^{n-1}}) & & & & \sin(\theta_{2^{n-1}}) \\ \hline \boxed{-\sin(\theta_1)} & & & & \boxed{\cos(\theta_1)} & & & \\ & -\sin(\theta_2) & & & & \cos(\theta_2) & & \\ & & \ddots & & & & \ddots & \\ & & & -\sin(\theta_{2^{n-1}}) & & & & \cos(\theta_{2^{n-1}}) \end{array}\right] \left[\begin{array}{c|c} W_1 & \\ \hline & W_2 \end{array}\right].$$

- $V_1, V_2, W_1, W_2 \in \mathrm{U}(2^{n-1})$
- The middle CS block is a **multi-controlled single-qubit unitary**:
  - $\sum_{x \in \{0,1\}^{n-1}} R_y(\theta_x) \otimes |x\rangle\langle x|$
  - Controlled by the last $n-1$ qubits, act on the 1st qubit

Recursive CS decomposition

- Apply CS decomposition to V_1, V_2, W_1, W_2 simultaneously:

$$V_1 = \begin{bmatrix} A_1 & \\ & A_2 \end{bmatrix} \cdot \begin{bmatrix} C_0 & S_0 \\ S_0 & -C_0 \end{bmatrix} \cdot \begin{bmatrix} B_1 & \\ & B_2 \end{bmatrix} \text{ and } V_2 = \begin{bmatrix} A_3 & \\ & A_4 \end{bmatrix} \cdot \begin{bmatrix} C_1 & S_1 \\ S_1 & -C_1 \end{bmatrix} \cdot \begin{bmatrix} B_3 & \\ & B_4 \end{bmatrix}$$

- Put together:

$$\begin{bmatrix} V_1 & \\ & V_2 \end{bmatrix} = \begin{bmatrix} A_1 & & & \\ & A_2 & & \\ & & A_3 & \\ & & & A_4 \end{bmatrix} \cdot \begin{bmatrix} \boxed{\begin{matrix} C_0 & S_0 \\ S_0 & -C_0 \end{matrix}} & \\ & \boxed{\begin{matrix} C_1 & S_1 \\ S_1 & -C_1 \end{matrix}} \end{bmatrix} \cdot \begin{bmatrix} B_1 & & & \\ & B_2 & & \\ & & B_3 & \\ & & & B_4 \end{bmatrix}$$

- The A/B unitaries are **2-controlled $(n-2)$-qubit unitaries**
  - Controlled by the first two qubits
- The middle unitary is still a **multi-controlled single-qubit unitary**. Why?
  - If 1st qubit is $|0\rangle$, apply CS_0 block to the rest
  - If 1st qubit is $|1\rangle$, apply CS_1 block to the rest
  - CS_0 and CS_1 blocks: act on 2nd qubit, controlled by the last $n-2$ qubits

Multi-controlled k-qubit unitary

- With m recursions, the task of implementing U reduces to implementing
  - $1 + 2 + \cdots 2^{m-1}$ multi-controlled single-qubit unitaries
  - 2^m multi-controlled $(n - m)$-qubit unitaries
- **[GKW24]:** Any multi-controlled single-qubit unitary can be implemented to error ϵ with

$$O\left(\sqrt{2^n \log(1/\epsilon)} + \log(1/\epsilon)\right) \text{ T gates and ancillae}$$

- **Our result:** Write $L = k + \log(1/\epsilon)$. Any multi-controlled k-qubit unitary can be implemented to error ϵ with

$$O\left(\sqrt{2^{n+k} \cdot L} + 4^k \cdot L\right) \text{ T gates and ancillae}$$

- **Overall:** with $k = n - m$, the minimal T-count of $\tilde{O}(2^{4n/3})$ happens at $k \approx n/3$
  - The associated ancilla-count is $\tilde{O}(2^{2n/3})$

Summary & outlook

- We study the T-count of implementing an arbitrary n-qubit unitary
- Previously the best T-count is $O(2^{1.5n})$, achieved by two different approaches
- We break the "3/2" barrier and improve it to "4/3"
- Our approach:
  - recursively apply CS decomposition to break an unstructured unitary into simpler, more structured pieces: multi-controlled k-qubit unitaries
- How to achieve the conjectured scaling of $O(2^n)$?
  - How to make more use of significantly more ancillae?